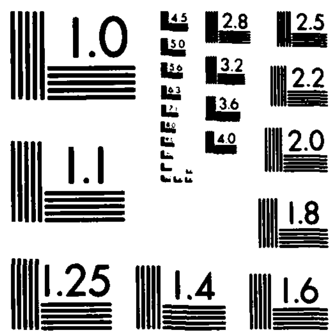
AD-A186 438   PROGRAMMABLE REAL-TIME INCOHERENT MATRIX-MULTIPLIER FOR OPTICAL PROCESSING(U) HUGHES RESEARCH LABS MALIBU CA Y OWECHKO JUL 87 HAC-REF. G1234 N00014-86-C-0811   1/1

UNCLASSIFIED   F/G 20/6   NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

1111 MB

25

HAC REF. G1234

(12)

DTIC FILE COPY

AD-A186 438

PROGRAMMABLE REAL-TIME INCOHERENT MATRIX-MULTIPLIER FOR OPTICAL PROCESSING

Yuri Owechko

Hughes Research Laboratories
3011 Malibu Canyon Road
Malibu, CA 90265

July 1987

N00014-86-C-0811
Final Report
September 1986 through April 1987

Monitored by :
OFFICE OF NAVAL RESEARCH
495 Summer Street
Boston, MA 02210-2109

87 8 11 071

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION
Unclassified

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE

3. DISTRIBUTION / AVAILABILITY OF REPORT

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
Hughes Research Laboratories

6b. OFFICE SYMBOL *(If applicable)*

7a. NAME OF MONITORING ORGANIZATION
Office of Naval Research

6c. ADDRESS *(City, State, and ZIP Code)*
3011 Malibu Canyon Road
Malibu, CA 90265

7b. ADDRESS *(City, State, and ZIP Code)*
495 Summer St.
Boston, MA 02210-2109

8a. NAME OF FUNDING / SPONSORING ORGANIZATION

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State, and ZIP Code)*

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*
Programmable Real-Time Incoherent Matrix-Multiplier for Optical Processing (Uncl)

12. PERSONAL AUTHOR(S)
Yuri Owechko

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM 9/86 TO 4/87

14. DATE OF REPORT *(Year, Month, Day)*
July 1987

15. PAGE COUNT
33

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Optical Computing
Ambiguity Function
Optical Signal Processing

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

In this Final Report, the Programmable Real-time Incoherent Matrix Multiplier for Optical Processing (PRIMO), which is based on outer-product decomposition, is described. PRIMO is a versatile optical processor which can multiply two NxN matrices in N clock cycles. In addition to matrix multiplication, PRIMO can perform such signal processing functions as correlation, convolution, 2-D Fourier transform, calculation of the cross-ambiguity function for both sliding and fixed windows (dynamic and static signals), matrix inversion, and histogram generation.

Special attention is paid to the optimum utilization of PRIMO algorithms for compensation of modulator and detector nonuniformities. For example, it is shown that an algorithm originally developed to represent bipolar and coupler numbers can also be utilized to mitigate modulator and detector bias nonuniformities. Optimum operating points for maximum dynamic range and bias nonuniformity compensation are derived.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION

22a. NAME OF RESPONSIBLE INDIVIDUAL

22b. TELEPHONE *(Include Area Code)*

22c. OFFICE SYMBOL

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

SECTION 1

INTRODUCTION

The objectives of this program were to study some of the practical issues involved in developing an optical processor based in part on outer-product multiplication of matrices, especially with regard to detector nonuniformities. Such a processor would be programmable, compact, fast, and would have many applications in the processing of image and radar data, solution of large systems of differential equations, beam forming and nulling, and in many other electronic warfare applications as well.

The propagation properties of light can be utilized for signal processing and computing with large advantages over electronic computers in terms of parallel operation. Many analog optical processing systems have been proposed and implemented in the past in order to perform useful linear signal processing operations (i.e., correlation, convolution, Fourier transform, etc.) on both one-dimensional (1-D) signals (usually in time) and on two-dimensional (2-D) signals (in space, time, or frequency), such as images or synthetic aperture radar data. By utilizing the parallelism of optics, such processors have, in many cases, achieved a large data throughput advantage over digital computers. In most cases, they require coherent light with all of its associated disadvantages such as poor signal-to-noise ratio and, in some cases, interferometric tolerance requirements.

Much work has also been reported on various optical vector-matrix and matrix-matrix multipliers for optical computing. An advantage of these matrix multipliers is that, since their operation does not depend on the coherence of the light source, incoherent light can be used (except for schemes utilizing acousto-optics). Linear operations on signals, such as correlation, can be expressed in terms of the algebraic manipulation and multiplication of matrices. Therefore, optical

matrix-matrix multipliers can also be utilized for signal processing as well as for optical computing functions such as matrix inversion. Such matrix processors will have an improved signal-to-noise ratio compared to analog processors which utilize coherent light, while still maintaining a high degree of parallelism.

At Hughes Research Laboratories (HRL), we have developed a method for performing optical matrix-matrix multiplication based on the outer-product decomposition of matrices. This method overcomes one of the main drawbacks of previously proposed optical matrix multipliers; the need for a 2-D spatial light modulator (SLM). By expressing the product of two matrices as a sum of matrices, each of which is the outer-product of a row of one matrix and a column of the other, 1-D SLMs can be used. This greatly reduces the hardware requirements since currently available 2-D SLMs cannot operate at the high frame rates required and are not, in general, as highly developed as 1-D SLMs. The addressing requirements are also reduced as compared to an electrically addressed 2-D SLM.

An advantage of this implementation, as opposed to acousto-optic implementations of outer-product processors, is complete control of data clocking rates. Data can be shifted through the processor at various rates without regard to acoustic velocities, providing flexibility in system design. Also, our approach does not require the use of coherent light and lenses for processing, thus reducing size and alignment requirements.

In this Final Report, the Programmable Real-time Incoherent Matrix Multiplier for Optical Processing (PRIMO), which is based on outer-product decomposition, is described. PRIMO is a versatile optical processor which can multiply two NxN matrices in N clock cycles. In addition to matrix multiplication, PRIMO can perform such signal processing functions as correlation, convolution, 2-D Fourier transform, calculation of the cross-ambiguity function for both sliding and fixed windows (dynamic and static signals), matrix inversion, and histogram generation.

Special attention is paid to the optimum utilization of PRIMO algorithms for compensation of modulator and detector nonuniformities. For example, it is shown that an algorithm originally developed to represent bipolar and coupler numbers can also be utilized to mitigate modulator and detector bias nonuniformities. Optimum operating points for maximum dynamic range and bias nonuniformity compensation are derived.

SECTION 2

TECHNICAL DESCRIPTION OF PRIMO

2.1 MATRIX MULTIPLICATION AND THE FOURIER TRANSFORM

The basic architecture of PRIMO is illustrated in Figures 1 and 2. It is best understood by analyzing its operation for matrix multiplication. PRIMO utilizes the principle of outer product decomposition for optical matrix multiplication. The product matrix C of two matrices B and A is given by

$$C = BA \quad , \tag{1}$$

where the ij-th element of C is given by the inner product between the i-th row vector of B and the j-th column vector of A:

$$c_{ij} = \sum_m b_{im} a_{mj} \quad . \tag{2}$$

However, C can also be written as a sum of matrices, each of which is the outer product between a column vector of B and the corresponding row vector of A. The principle behind an outer product matrix multiplier is to sequentially feed the rows of matrix B into a 1-D SLM and the corresponding columns of matrix A into another 1-D SLM which is orthogonal to the first SLM. The device is entirely edge-addressed. The transmission of the two crossed 1-D SLMs during the nth clock cycle is given by the outer product of the nth row of B and the nth column of A. The transmitted light falls on a 2-D accumulator detector array and summed to form the product matrix C. The multiplication of two NxN matrices, which requires N^3 multiplications, is performed in N clock cycles.

Figure 1 shows the two matrices, A and B, being fed into PRIMO (row and column at a time, respectively). The two orthogonally oriented 1-D SLMs consist of linear electrodes deposited on thin electro-optic crystal slices. (Polarizers that

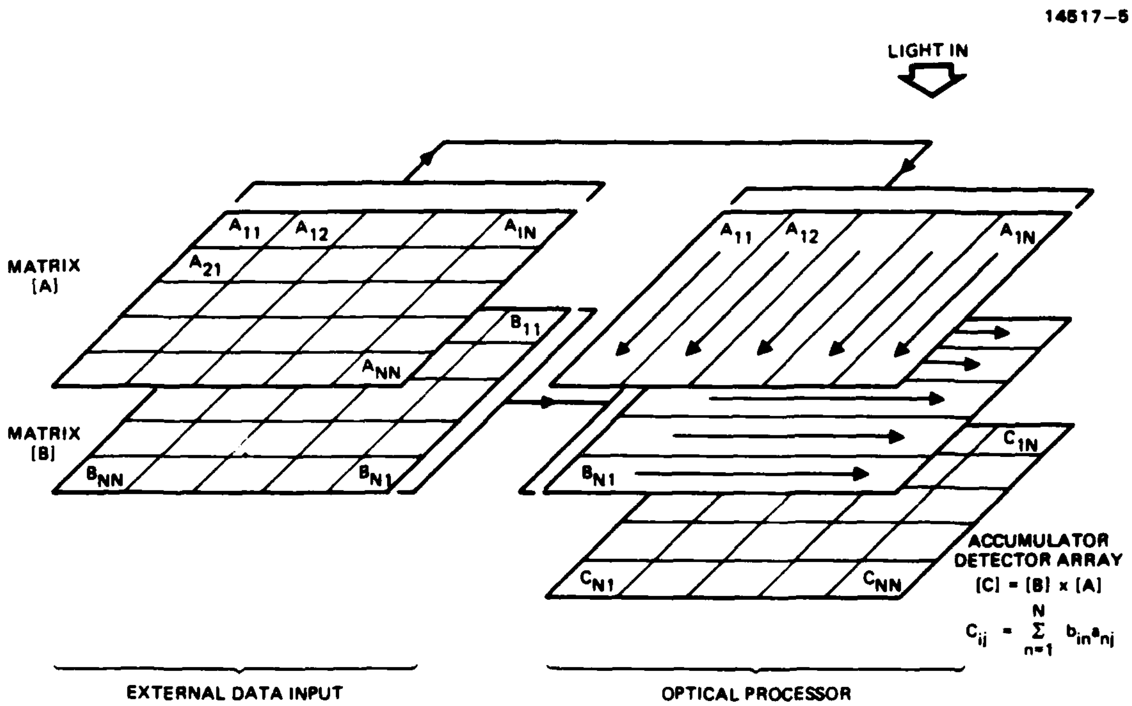


Figure 1. Matrix multiplication using outer product decomposition.

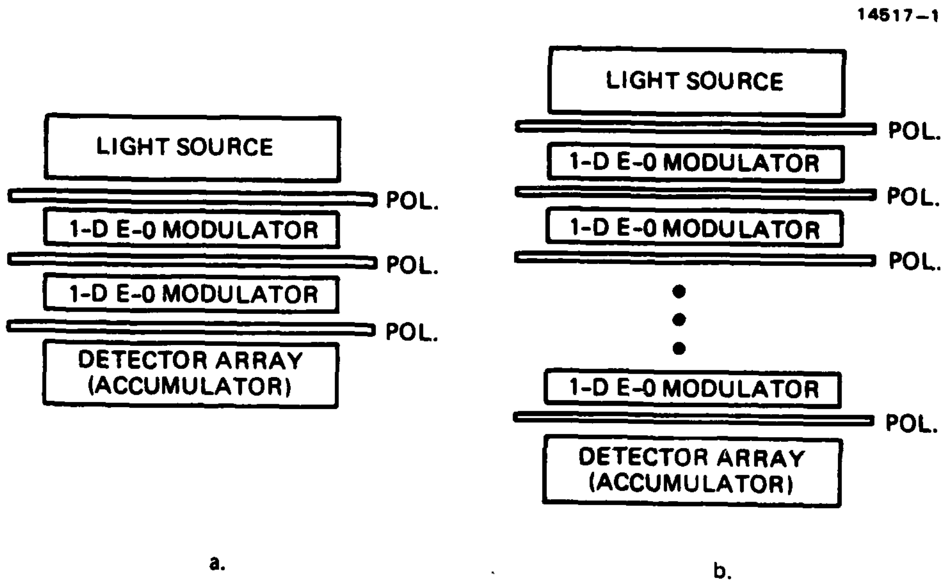


Figure 2. PRIMO architecture.

are located between the electro-optic crystals have been omitted from Figure 1 for the sake of clarity.) Since the electrodes in each layer are linear striped, either the transverse or longitudinal electro-optic effect can be used. During the nth clock cycle, light incident on PRIMO is modulated in one direction by the nth row of A and in the orthogonal direction by the nth column of B, forming the nth outer product matrix at the accumulator detector array, the sum of which is the product matrix C.

Many electro-optic crystal layers can be stacked together as shown in Figure 2. Figure 2(a) shows the basic device configuration for matrix-matrix multiplication. By making the layers thin, no lenses are required between the layers and an extended incoherent light source can be used. Figure 2(b) shows a multilayer programmable stack of 1-D electro-optic modulators which can be used for cascaded operations and for more complicated operations such as generation of the cross-ambiguity function between two signals, which will be described below.

The Fourier transform of 2-D data can be calculated by utilizing the basic configuration of Figure 1 and Figure 2(a) because Fourier transformation is a special case of matrix-matrix multiplication. For example, if a 1-D Fourier transform of 2-D data is desired, the 2-D data are placed in matrix B and the corresponding Fourier exponential terms in matrix A of Figure 1. The processor is then stepped through the sequence described above for matrix multiplication. The product matrix C in the accumulator is then the 1-D Fourier transform of matrix B. If a 2-D Fourier transform is required, then the previously calculated C matrix values must be transferred back to the B matrix and the processor is stepped through another sequence with a different set of Fourier exponential terms in the A matrix which now correspond to a 1-D Fourier transform in the orthogonal direction. The final result in the accumulator after 2N clock cycles will be the 2-D Fourier transform of the 2-D input data, assuming the input array is NxN.

2.2 CORRELATION AND THE CROSS-AMBIGUITY FUNCTION

An interesting signal processing operation, important in radar, for example, is the calculation of the sliding window cross-ambiguity function described by the equation

$$A(\nu,\tau) = \int_{o}^{T} G(t)F(t-\tau)\exp(i2\pi\nu t)dt \quad , \tag{3}$$

where F(t) is a continuously running signal and G(t) is a finite reference template of length T. Correlation is a special case of Eq. (3) for $\nu = 0$.

The PRIMO architecture for calculating the sliding window cross-ambiguity function is shown in Figure 3. The Fourier exponential terms are located in matrix E, the template function G is continuously applied to one electro-optic modulator layer as shown, and the continuously running signal F is input into an electro-optic modulator layer that has had its rows shorted across the entire plane. This layer can be eliminated by using a pulsed light source modulated by F, such as an LED or laser diode. A further advantage of using such a source is that the detector plane can be easily shuttered during clocking of data from one cell to the next. The PRIMO output A_{ln} is given by

$$A_{ln} = \sum_{m=o}^{M-1} f_{n-m}g_{m}\exp(i2\pi l(n-m)/M) \quad . \tag{4}$$

The indices n, m, and l correspond to delay time, τ, time t, and frequency, ν, respectively. Equation (4) is equivalent to Eq. (3) except that f_{n-m} is reversed in time. This is not a problem so long as transconductance, g_m, is also reversed. (Convolution instead of correlation results if g_m is not reversed.) The objective is to correlate the most recent M samples of the F function (weighted by the Fourier exponential)

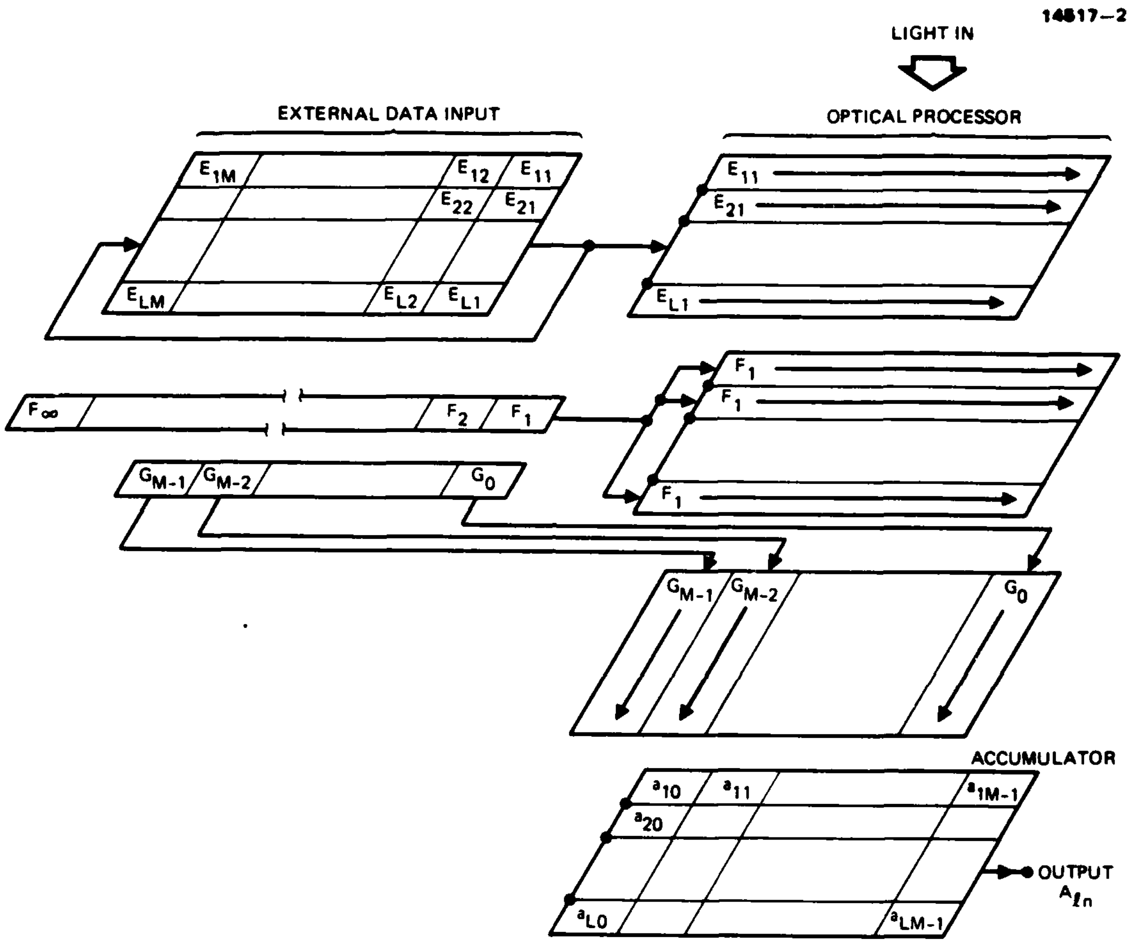


Figure 3. Calculation of the sliding window cross-ambiguity function.

with the fixed G template. The summation is carried on the product of the most recent M samples of the F function and the M samples of the G function. Since the E terms of the Fourier exponential matrix are periodic in time with period M, they are recirculated. In each clock period a new update for the A function is extracted.

The a_{ij} terms marked on the accumulator in Figure 3 have a different meaning from the C_{ij} terms of Figure 1. The a_{ij} terms are partial sums (intermediate results), and at each clock cycle are shifted one cell to the right and a new term added. They are gradually built up to the full value of M terms and then output as A_{ij}; therefore, $a_{i,M-1} = A_{in}$. This feature is a result of the sliding window nature of this particular architecture and results in the real-time calculation of the cross-ambiguity function for continuously running 1-D input signals. However, fixed window correlations and ambiguity functions for static or fixed input data can also be easily implemented using the PRIMO approach.

The general algorithm described above can be used to implement any triple product form besides the ambiguity function. Triple correlation or the Wigner distribution can be calculated as well with a high degree of parallelism.

2.3 FADDEEV ALGORITHM

The Faddeev algorithm calculates the matrix form $CA^{-1}B+D$ from given matrix inputs A, B, C, and D. Important special uses are matrix inversion and multiplication, the solution of linear equations and least square problems. This section describes a PRIMO architecture for the optical implementation of the Faddeev algorithm by means of the Gaussian elimination or condensation technique. The operations in PRIMO are done in parallel and the algorithm is programmable in the sense that any of its special cases can be implemented readily.

A summary of the algorithm is shown in Figure 4. Four given matrices A, B, C, and D are placed in a four quadrant field as shown. The matrix A is multiplied by matrix W and the result is added to the third quadrant field (-C); the same is done to the second quadrant field (B) and the fourth quadrant field (D). A Gaussian elimination procedure (explained later) is used to find W, so that WA-C = 0 or $W = CA^{-1}$. In this case the third quadrant field vanishes and in the fourth quadrant one obtains

$$CA^{-1}B+D \quad ,$$

which includes matrix multiplications, inversion, and addition as special cases.

In Figure 5 some particular results obtainable with the Faddeev algorithm are shown. In the left column are shown the input matrices that are placed in the four quadrants of the field. By using Gaussian elimination, the outputs shown in the right column are obtained. The output will appear in the fourth quadrant of the field. The top entry is the general case discussed in the previous figure. Assuming A = 1 (unity matrix), C = 1 and D = 0, matrix inversion and multiplication result. For A = 1 and D = 0, the matrix product of CB results, and for B = C = 1 and D = 0, matrix inversion is obtained. It is important to note that one obtains the different functions merely by changing the input data, not the system architecture; therefore, this system is highly programmable.

Using the well-known Gaussian elimination technique and treating the four matrices in the four quadrants as one matrix of order 2N, one calculates terms in a new matrix with the following formula:

$$X_{nm}^{new} = X_{nm}^{old} - \frac{X_{n1}^{old} \cdot X_{1m}^{old}}{X_{11}^{old}} \quad . \tag{5}$$

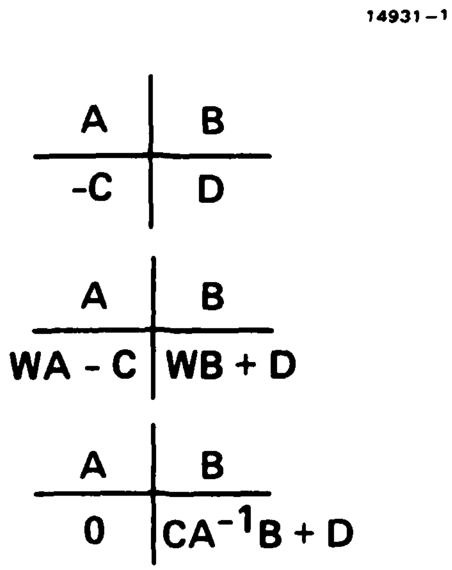


Figure 4. The Faddeev algorithm.

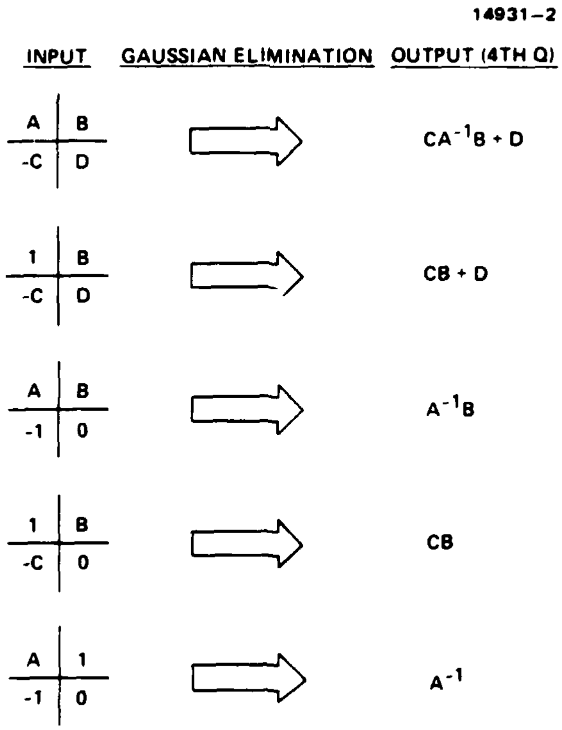


Figure 5. Special cases of the Faddeev algorithm.

All the terms in the top row and left column of the "new" matrix become zero. (This is easy to verify by substituting n = 1 or m = 1 or n = m = 1 in the above expression.) Therefore the "new" matrix is reduced to order 2N-1. If this procedure is repeated N-1 times more, a matrix of order N given by the expression $CA^{-1}B+D$ results. If during this procedure an upper left corner term (X_{11}^{old}) is zero, "partial pivoting," is done, which means to exchange the first row with any other nonzero first term row, while at the same time these two rows will also be exchanged in the final (output) matrix. This entire procedure is familiar as a method of solving a set of linear equations.

In Figure 6 the Faddeev algorithm and the Gaussian elimination procedure are applied to the matrix inversion problem. Assuming for the input data B = 1 in the first quadrant, C = 1 in the third quadrant, and D = 0 in the fourth quadrant, one obtains A^{-1}. For example, let

$$A = \begin{bmatrix} 5 & 3 \\ 9 & 4 \end{bmatrix}$$

be a given 2x2 matrix. The 4x4 input extended matrix is shown in the lower left corner of the diagram. Using Eq. (5), the matrix shown in the center of the diagram is calculated term by term. The first row and the first column in the new matrix are zeros. Applying Eq. (5) again to this 3x3 matrix, one obtains the matrix shown in the right of the diagram. This final matrix is 2x2 and it is the desired result A^{-1}. It is important to note that the variable information at each step is only of size NxN. Therefore, one could use an NxN system and shift the information one position north and west at each step. In addition, one would have to temporarily store the uppermost row and the left most column of the "old" matrix to calculate the terms of the "new" matrix at each step.

14931–4

MATRIX INVERSION EXAMPLE

$$CA^{-1}B + D \qquad \begin{array}{c|c} A & 1 \\ \hline -1 & 0 \end{array} \qquad X^{NEW}_{nm} = X^{OLD}_{nm} - \frac{X^{OLD}_{n1} \; X^{OLD}_{1m}}{X^{OLD}_{11}}$$

$$\begin{array}{cccc} 5 & 3 & 1 & 0 \\ 9 & 4 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{array} \qquad \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & -1.4 & -1.8 & 1 \\ 0 & 0.6 & 0.2 & 0 \\ 0 & -1 & 0 & 0 \end{array} \qquad \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -0.57 & 0.43 \\ 0 & 0 & 1.29 & -0.71 \end{array}$$

Figure 6. Matrix inversion using the Faddeev algorithm.

In Figure 7 the implementation of matrix inversion using Faddeev algorithm plus Gaussian elimination in the PRIMO system is shown. At the bottom of the system there is an (N+1)x(N+1) accumulator. The extra row and column (shaded in the diagram) are used to store the uppermost row and the left most column of the "old" matrix. The matrix to be inverted is loaded into the accumulator in the unshaded area and is stepped one north and one west. In addition to the accumulator there are three active EO layers. The upper one (EO 1) is fed by the inverse of the uppermost left term of the "old" matrix ($1/X_{11}^{old}$).

The important advantage of the Faddeev-Gaussian procedure is that this divisor is constant for the whole array in a given step of the transformation. This enables one to calculate the inverse of this term (shown in the diagram as 1/X) in a serial electronic circuit and then use the calculated value to multiply the whole area using a "shorted" EO layer. The EO 2 is fed by the remainder of the terms of the left most row of the "old" matrix and -1 as shown in Figure 7. Similarly, the EO 3 is fed by the remainder of the terms of the uppermost row of the "old" matrix and 1. The result of the triple multiplication $(X_{n1}^{old})(X_{1m}^{old})/X_{11}^{old}$ is subtracted from the values stored in the accumulator. The information is shifted one step north and one step west and the triple multiplication with the subtraction is repeated. This procedure is repeated N times. At the end, the inverted matrix is stored in the unshaded area of the accumulator. The "zero" registers, shown next to the accumulator, will not exist in a practical system; they are shown here for display purposes only. In a practical system the accumulator will be designed in such a way that, when shifted north and west, zeros will enter to the bottom row and the right most column. The pivoting circuit (not shown in the diagram) will be activated each time zero appears in the uppermost left pixel and the control unit will keep track of these pivotings.

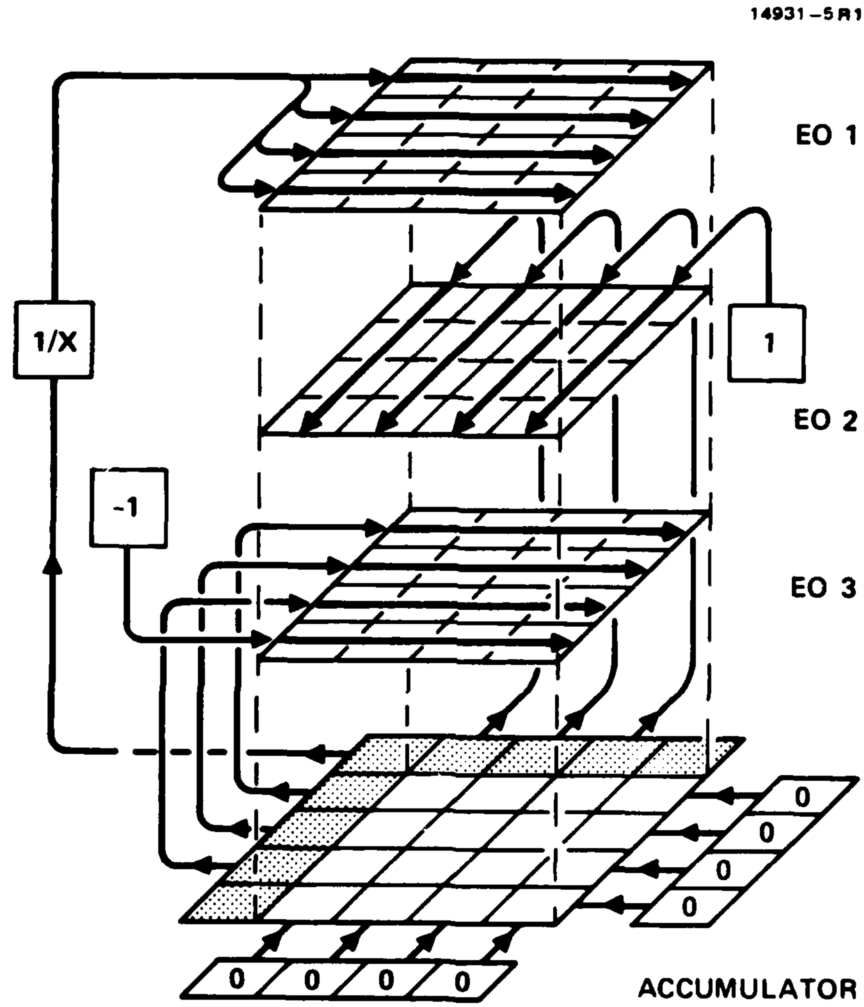


Figure 7. PRIMO implementation of matrix inversion.

In Figure 8 the implementation of the complete Faddeev algorithm plus Gaussian elimination scheme for the PRIMO system is shown. The output of this system will be $CA^{-1}B+D$. Assume that the accumulator is of size 2Nx2N, the same as the extended input matrix, and that it will be loaded into the accumulator. The procedure is the same as in the case of matrix inversion. First one multiplies and subtracts. Then the accumulator is shifted one step north and one west; again multiply and subtract. This procedure is repeated N times. The result $CA^{-1}B+D$ appears in the upper left corner of the accumulator.

Negative and complex numbers can be handled as will be shown in a subsequent section.

2.4 HISTOGRAM

The generation of histograms of 1- or 2-D signals is an important operation in signal and image processing. The calculation of the histogram of an NxN pixel image using a serial computer requires nxNxN operations (where n is the number of levels) and is very time consuming. The level of each pixel must be compared to the n set levels. As shown in Figure 9, the parallelism and edge addressing capability of PRIMO can be used to generate the histogram of an NxN image in N clock cycles.

Two crossed, 1-D electrooptic (EO) modulator layers are shown schematically at the bottom of Figure 9 with no polarizer between them. The two EO layers are situated between crossed polarizers which results in the addition or subtraction of signals applied to the two layers, depending on their relative polarities. This effect is utilized as an n level comparator by positioning a set of nxN zero or null detectors underneath the EO modulators. A zero detector is activated when the voltages applied to the two EO modulators are equal. The 2-D NxN signal or image is applied one line at a time to the top EO layer. The bottom EO layer is addressed by a set of n fixed (but programmable) voltages which represents the n signal levels into

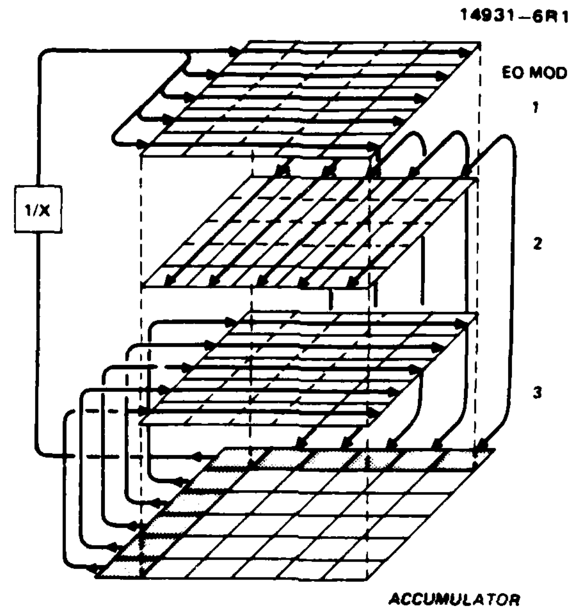


Figure 8. PRIMO implementation of the general Faddeev algorithm.

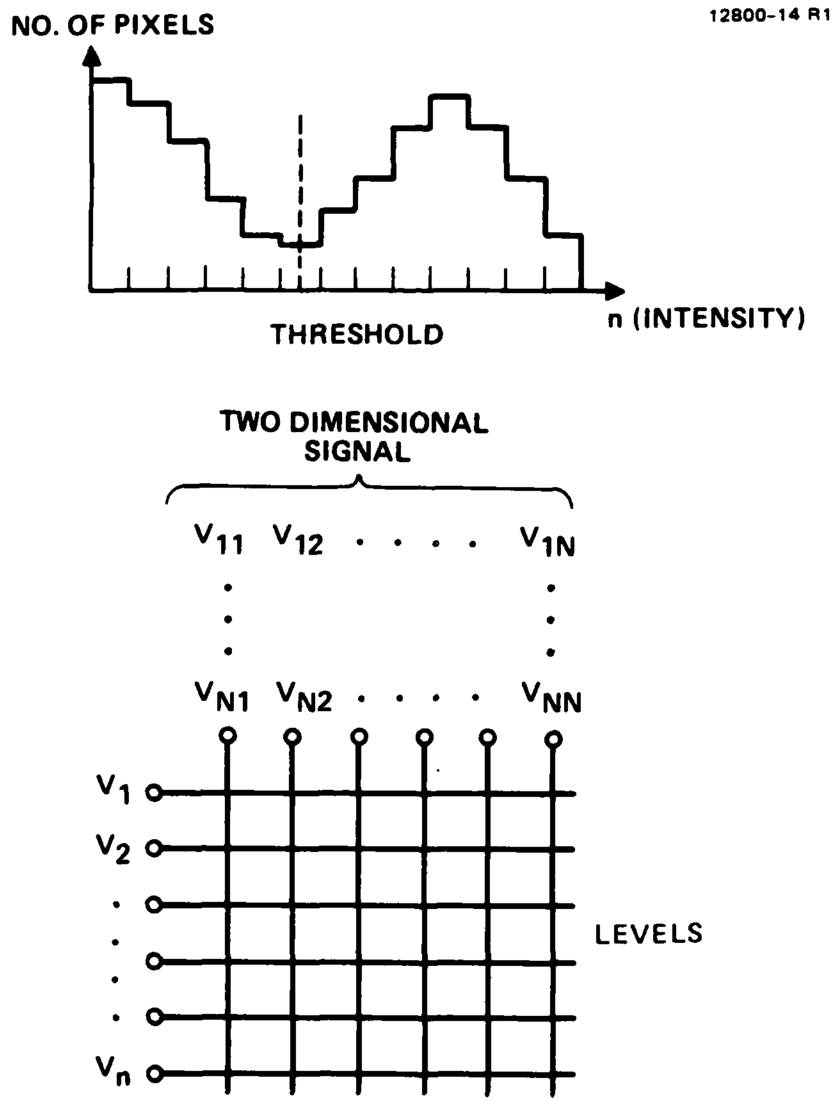


Figure 9. Histogram generation using PRIMO.

which the pixels of the input are to be sorted. Each zero detector feeds one of n counters which keeps track of the number of pixels in each of the n levels. Each line of the input data is compared in parallel to the n signal levels. The histogram, therefore, is generated in N clock cycles.

2.5 BIPOLAR AND COMPLEX NUMBER REPRESENTATION

In incoherently illuminated optical processors, numbers are represented by light intensities which are nonnegative quantities. Most operations, however, involve bipolar and often complex numbers. A bias-based time and space multiplexed method for representing bipolar and complex numbers and which linearizes the modulator-detector response is described in this section.

A shortcoming common to most optical matrix multiplication techniques is the square law detector nonlinearity. Modulators based on electro-optic crystals modulate light amplitude linearly in response to an applied voltage (for voltages that are small compared to the half-wave voltage), while most detectors respond to light intensity. The detector output is therefore proportional to the square of the applied voltages. For example, the combined amplitude transmission of two stacked EO modulator layers with polarizers between the layers is given by

$$
\begin{aligned}
t &= t_a t_b \\
&= \sin(\Delta_a + \phi_a)\sin(\Delta_b + \phi_b) \\
&\simeq (\Delta_a + \phi_a)(\Delta_b + \phi_b) \quad ,
\end{aligned}
$$

where ϕ_x is the birefringent phase shift induced by voltage X and Δ_x is a constant bias, which may be the result of crystal birefringence or a constant voltage bias. It is assumed above that Δ_x and ϕ_x are small enough to neglect the sine nonlinearity. The detector response is proportional to $|t|^2$, which is clearly not proportional to the desired product, $\phi_a\phi_b$.

The square law detection nonlinearity can be eliminated while simultaneously allowing the representation of bipolar numbers by introducing a bias and sequencing the data in a special way. The bias-based method for linear bipolar number multiplication is illustrated in Figure 10. The input data, ϕ_x, are added to the constant bias terms, Δ_x. The bipolar input data, ϕ_x, are multiplied by +1 or -1, as shown, regardless of their polarity. Including the sine nonlinearity resulting from the transfer function of the electro-optic modulators, the contents of the plus and minus cells of the integrating detector are proportional to

$$d_+ = [\sin(\Delta_a + \phi_a)\sin(\Delta_b + \phi_b)]^2 + [\sin(\Delta_a - \phi_a)\sin(\Delta_b - \phi_b)]^2$$

$$d_- = [\sin(\Delta_a + \phi_a)\sin(\Delta_b - \phi_b)]^2 + [\sin(\Delta_a - \phi_a)\sin(\Delta_b + \phi_b)]^2 \quad .$$

The bipolar electrical output of the difference amplifier is given by the difference between the contents of the plus and minus cells of the detector. (Alternatively, the difference could be taken by convolving the output plane with a sine function of width equal to the two detector cells.) By using simple trigonometric identities, it can be shown that the amplifier output is proportional to

$$d = d_+ - d_- = \sin(2\Delta_a)\sin(2\Delta_b)\sin(2\phi_a)\sin(2\phi_b) \quad .$$

For input data voltages that are small compared with the electro-optic crystal half-wave voltage, the output voltage is linearly proportional to the input data, ϕ_a and ϕ_b, and has zero bias. This technique removes the square law detection nonlinearity because it eliminates all of the even order terms in the power series expansion in ϕ about Δ of the modulator transmittance, leaving only the odd order terms. An interesting point that will be discussed further in the next section is that the size of the

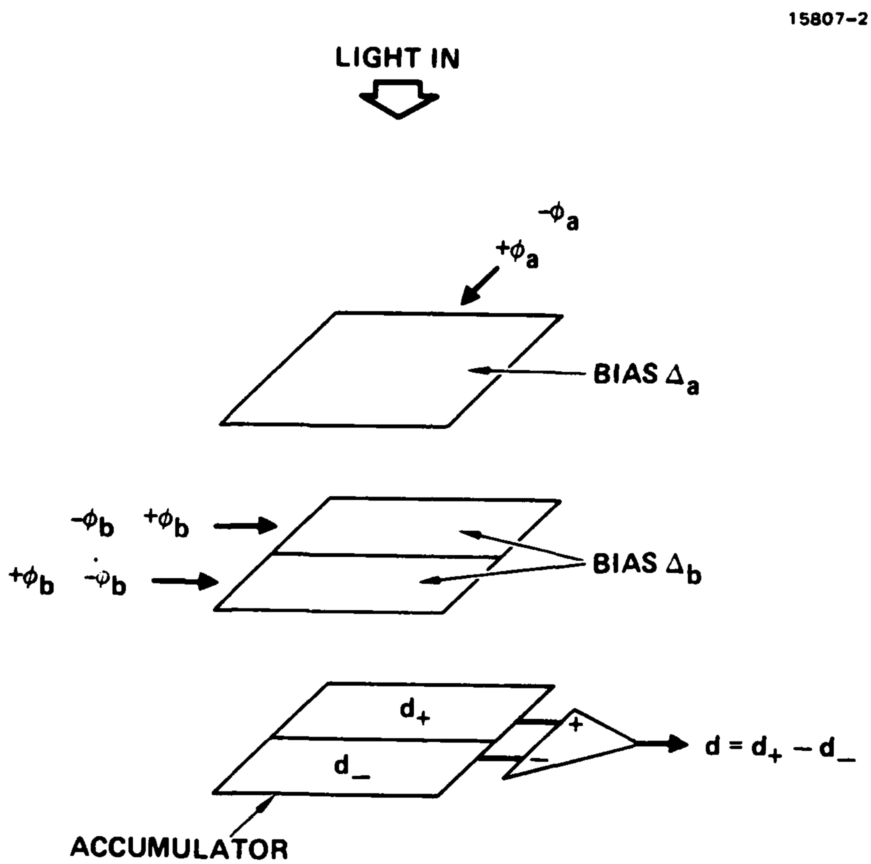


Figure 10. Bias-based method for bipolar multiplication.

bias has no effect on the linearity for linear electrooptic materials. The sine nonlinearity remains, but is small for input data voltages that are small compared with the half-wave voltage. All sources of bias are compensated to the extent that the bias is uniform between adjacent plus and minus cells. Detector dark current bias is compensated, as well as optical bias arising from EO crystal birefringence or incomplete polarizer extinction. This mitigates some of the detector noise sources and increases the effective dynamic range of the processor, as described further in the next section.

Since the bias-based method eliminates all even order terms in the power series expansion of the modulator transfer function, it will also work without any changes for quadratic as well as linear EO materials. In quadratic materials, such as some forms of PLZT, the birefringent phase shift is proportional to the square of the applied voltage instead of the voltage itself. For quadratic materials, the bipolar detector output d obtained using the bias-based method is given by

$$d_+ = [\sin(\Delta_a + \phi_a)^2 \sin(\Delta_b + \phi_b)^2]^2 + [\sin(\Delta_a - \phi_a)^2 \sin(\Delta_b - \phi_b)^2]$$

$$d_- = [\sin(\Delta_a + \phi_a)^2 \sin(\Delta_b - \phi_b)^2]^2 + [\sin(\Delta_a - \phi_a)^2 \sin(\Delta_b + \phi_b)^2]$$

$$\begin{aligned} d &= d_+ - d_- \\ &= \underbrace{[\sin^2(\Delta_a + \phi_a)^2 - \sin^2(\Delta_a - \phi_a)^2]}_{\text{Term A}} \; \underbrace{[\sin^2(\Delta_b + \phi_b)^2 - \sin^2(\Delta_b - \phi_b)^2]}_{\text{Term B}} \end{aligned}$$

The linearization of the PLZT modulator transfer function is illustrated in Figures 11 and 12. Figure 11 shows the unprocessed output of a quadratic PLZT modulator. The output

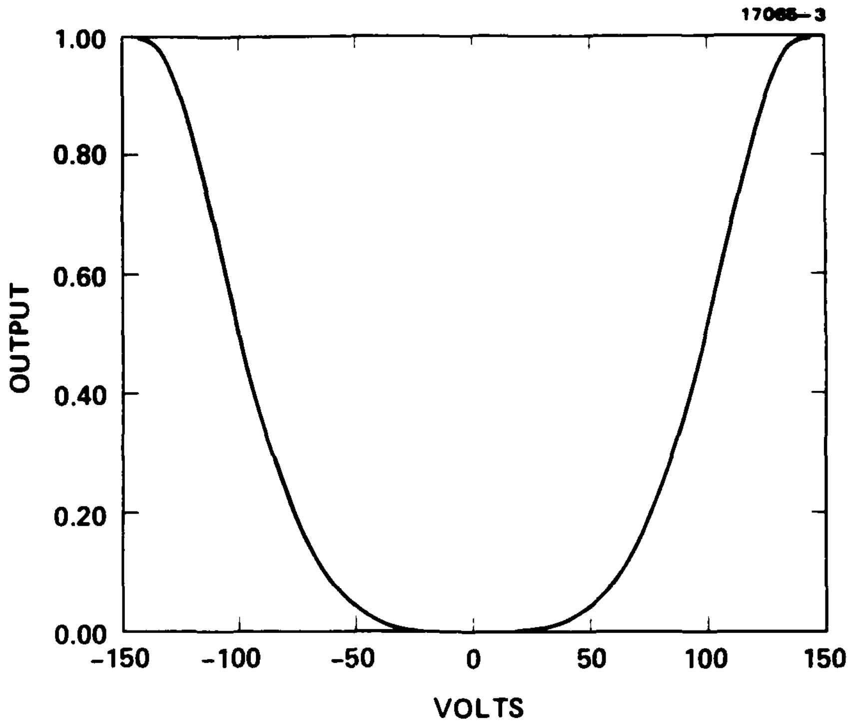


Figure 11. PLZT modulator transfer function (100 μm gaps).

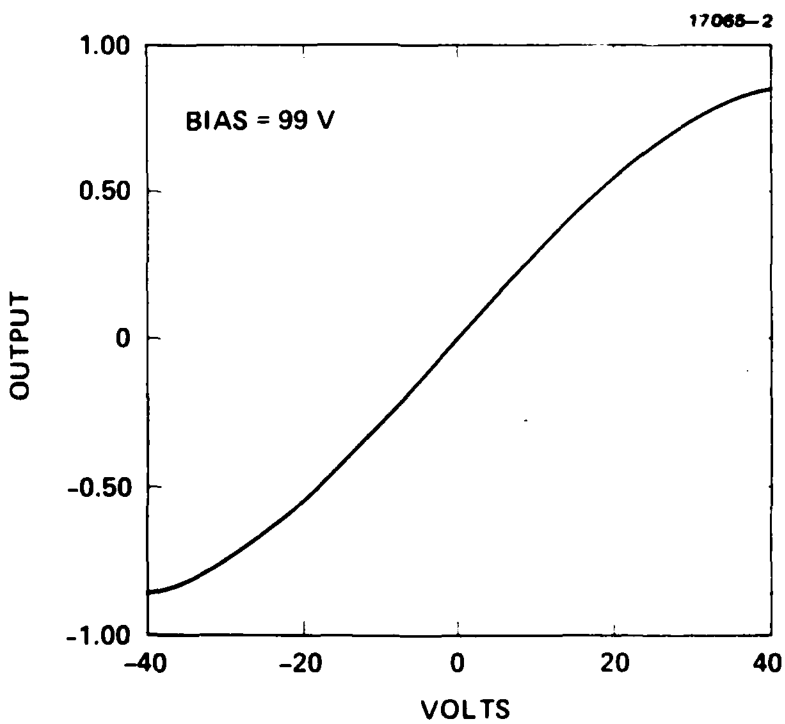


Figure 12. Linearized PLZT modulator transfer function (100 μm gaps).

linearized using the bias-based method is illustrated in Figure 12 where Term A from the above equation is plotted as a function of the signal ϕ_a. The output is quite linear for large variations in ϕ.

The control circuitry for the bias-based method is simple because the data input algorithm is independent of the polarity of the data. The data are sequenced without regard to their polarity.

A bias-based method for linear representation of complex multiplication is illustrated in Figure 13. It is a straightforward extension of bipolar multiplication. The real and imaginary parts of the data are represented as bipolar quantities. Upon readout, the output of the difference amplifier is first the imaginary part, d^i, and then the real part, d^r, of the product, given by

$$d^i = d_+ - d_- = 16\Delta_a\Delta_b(\phi_a^i\,\phi_b^r + \phi_a^r\,\phi_b^i)$$

$$d^r = d_+^r - d_-^r = 16\Delta_a\Delta_b(\phi_a^r\,\phi_b^r - \phi_a^i\,\phi_b^i),$$

in agreement with the definition of complex multiplication. The real and imaginary parts of the product are obtained directly in a convenient bipolar, linear form. The above equation assumes $\phi<<\pi/2$ so that the sine nonlinearity can be neglected.

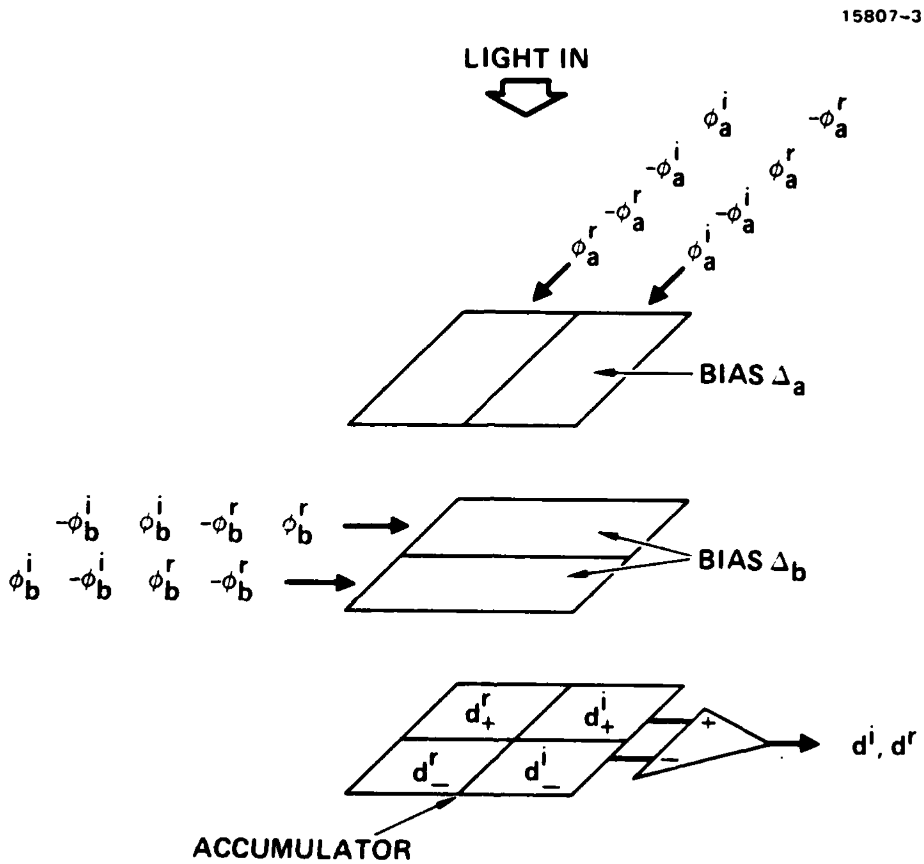


Figure 13. Bias-based method for complex multiplication.

SECTION 3

IMPACT OF PRIMO ALGORITHMS ON DETECTOR UNIFORMITY REQUIREMENTS

The detector, signal d, can be written as

$$d = S + \delta + n \quad ,$$

where S is the true signal, δ is a bias buildup from the time integration of both the detector dark current and residual light leakage through the EO modulators, and n is time-dependent noise modeled as a zero mean stochastic process with deviation σ. The dynamic range for conventional detection is given by

$$DR = \frac{d_{sat}}{\delta + \sigma} \quad ,$$

where d_{sat} is the saturated detector signal. Since the bias terms are subtracted out in the bias-based technique, the resultant theoretical dynamic range for the same conditions is given by

$$DR = \frac{d_{sat}}{\sigma} \quad .$$

The dynamic range is now limited by stochastic noise rather than by bias buildup. In practice, the effectiveness of the bias subtraction for the case of matrix multiplication will be limited by variations between adjacent detector "pixels."

The equation for the detector output for linear electrooptic materials is reproduced below.

$$d = d_+ - d_- = \sin(2\Delta_a)\sin(2\Delta_b)\sin(2\phi_a)\sin(2\phi_b) \quad .$$

Two relevant observations can be made regarding the above equation. First, the bias terms appear as multiplicative factors and, second, the bias terms are separable from the signal terms. Global variations (over distances greater than two modulator widths) in the bias due to spatial nonuniformities in detector dark current, therefore, manifest themselves as spatially varying inaccuracies in d. The separability of the bias and signal terms, however, can be exploited to compensate for the detector global bias variations. If Δ_a and Δ_b are set equal to $\pi/4$ radians, then the multiplicative bias terms will be biased at the maximum of the sine function where the derivative with respect to Δ is much less than 1. Small fractional variations in Δ will be transformed into much smaller fractional variations in d, thus providing partial compensation for spatial nonuniformities. It is fortuitous that the optimum value of the bias for nonuniformity compensation is also the optimum value for maximum signal gain.

The analogous situation for quadratic electrooptic materials is slightly more complicated. The relevant equation for d is reproduced below from the previous section:

$$d = d_+ - d_-$$

$$= \underbrace{[\sin^2(\Delta_a + \phi_a)^2 - \sin^2(\Delta_a - \phi_a)^2]}_{\text{Term A}} \; \underbrace{[\sin^2(\Delta_b + \phi_b)^2 - \sin^2(\Delta_b - \phi_b)^2]}_{\text{Term B}}$$

Term A can be further simplified:

$$\text{Term A} = \sin(4\Delta_a\phi_a) \; [\sin(2\Delta_a^2)\cos(2\phi_a^2) + \cos(2\Delta_a^2)\sin(2\phi_a^2)] \quad .$$

Term A is plotted in Figure 14 as a function of bias Δ_a, assuming ϕ_a is small ($\phi_a^2 = 0.01$ radians). In order to achieve the same nonuniformity compensation, a value of Δ_a must be found

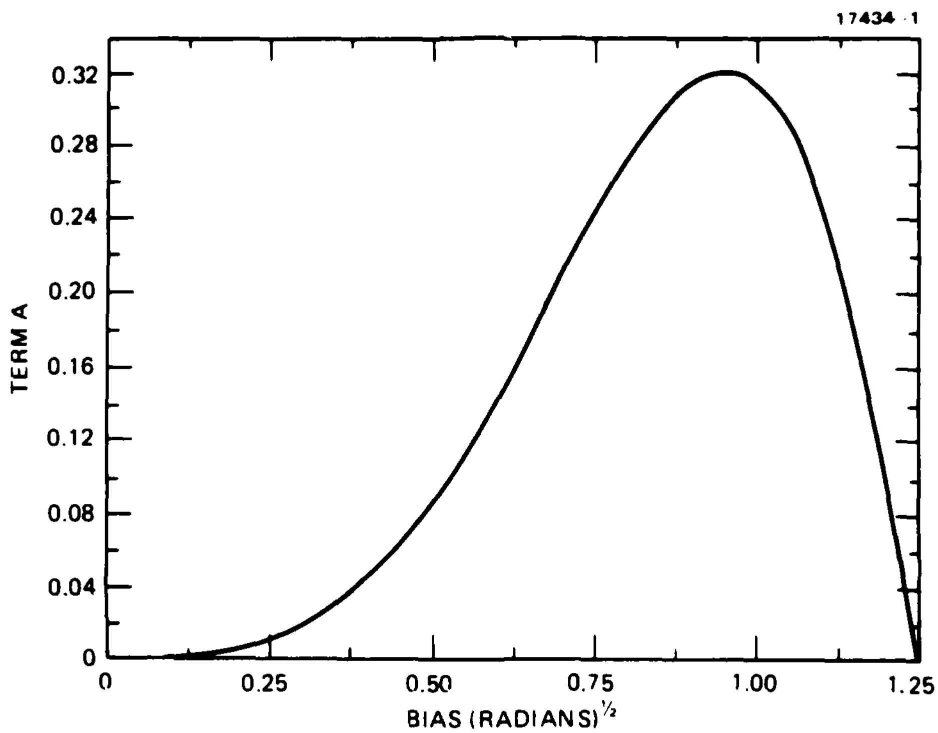


Figure 14. Dependence of detector output on bias for quadratic electrooptic effect using bipolar algorithm.

that minimizes the derivative of Term A with respect to the bias. From the figure it is clear that there is one candidate value that corresponds to the maximum of Term A where the derivative is zero. The derivative is plotted in Figure 15. It is apparent that there is an appreciable range of values about the zero derivative point where the derivative is less than one and bias variations can be compensated.

In applications involving correlation operations where data are shifted every clock cycle, the pixel variations will be averaged. Thus we expect the dynamic range performance for correlation type operations to be superior to that for matrix multiplication in that both local and global nonuniformities can be compensated.

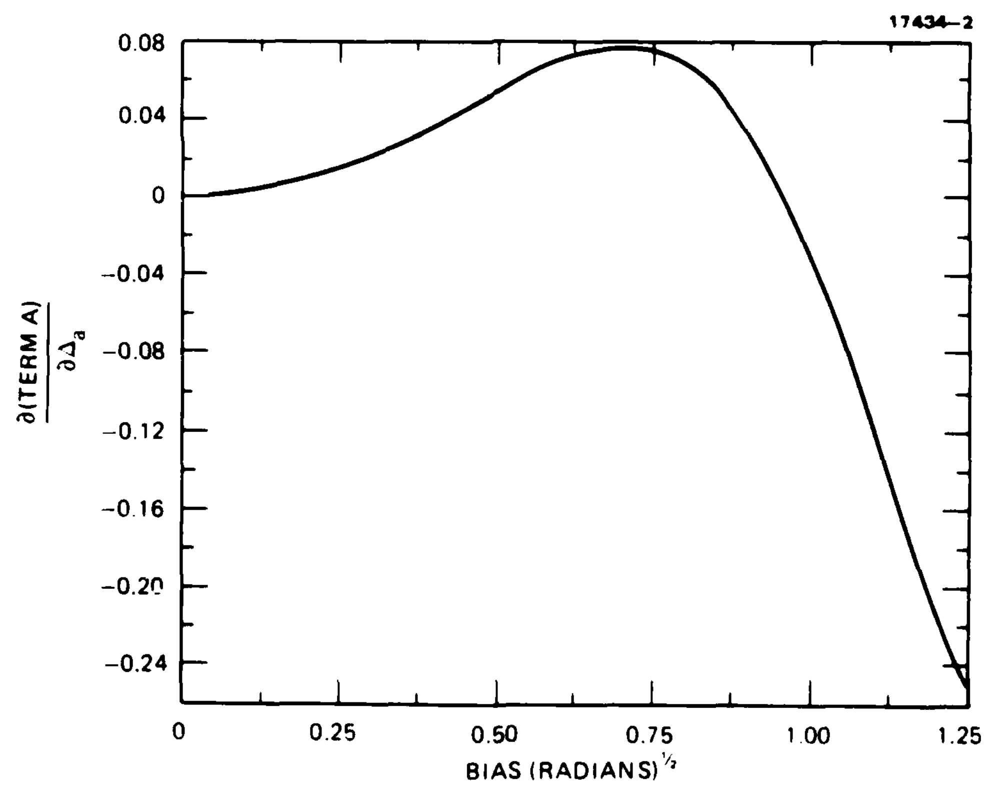


Figure 15. Derivative of Figure 14 with respect to the bias.

SECTION 4

SUMMARY

In this Final Report, the Programmable Real-time Incoherent Matrix Multiplier for Optical Processing (PRIMO), which is based on outer-product decomposition, was described. PRIMO is a versatile processor that can multiply two NxN matrices in N clock cycles. In addition to matrix multiplication, PRIMO can perform such signal processing functions as correlation, convolution, 2-D Fourier transform, calculation of the cross-ambiguity function for both sliding and fixed windows (dynamic and static signals), matrix inversion, and histogram generation.

It was shown that PRIMO algorithms developed for representing bipolar and complex numbers using incoherent light can also be utilized for compensation of modulator and detector nonuniformities. Both linear and quadratic electrooptic effect cases were analyzed and optimum values for bias levels were determined.

END

12-87

DTIC